



(12) 发明专利申请

(10) 申请公布号 CN 104065745 A

(43) 申请公布日 2014. 09. 24

(21) 申请号 201410321360. 1

(22) 申请日 2014. 07. 07

(71) 申请人 电子科技大学

地址 611731 四川省成都市高新区(西区)西
源大道 2006 号

(72) 发明人 罗光春 田玲 秦科 刘贵松
张蛟

(74) 专利代理机构 成都希盛知识产权代理有限
公司 51226

代理人 陈泽斌

(51) Int. Cl.

H04L 29/08 (2006. 01)

G06F 9/455 (2006. 01)

G06F 9/50 (2006. 01)

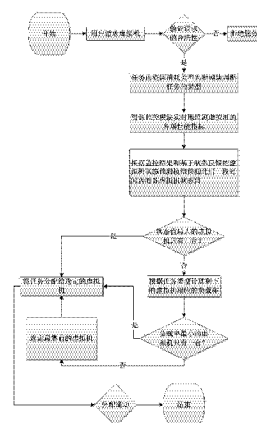
权利要求书2页 说明书8页 附图2页

(54) 发明名称

云计算动态资源调度系统和方法

(57) 摘要

本发明涉及云计算领域中的动态资源调度技术,其针对现有技术中的云计算资源分配和调度技术的不足之处,提出一种基于反馈和预测机制的云计算动态资源调度方法,能够在云计算环境中实现均衡使用各种计算机资源,实现在较小的开销下获得满意的负载均衡,提高系统调度的综合效率。本发明的方案概括为:在虚拟机运行的过程中实时地监测虚拟机的各项性能指标,在任务请求到来时,根据监测到的虚拟机当前的各项性能指标和基于状态反馈的虚拟机状态预测模型预测当前所有虚拟机在接下来的一小段时间内的运行状况,根据预测结果结合请求的任务类型选择出最适合的虚拟机,并分配给请求的任务。此外,本发明还公开了相应的资源调度系统,适用于动态资源调度。



1. 云计算动态资源调度系统,其特征在于,包括:

任务请求模块,用于在用户提交任务请求时,验证用户身份的合法性和请求资源的合法性,若验证未通过,则拒绝该任务请求;若验证通过,则将该任务请求发送给任务类型判断模块;

任务类型判断模块,用于在收到任务请求时,判断请求的任务类型,并将所述请求的任务类型发送给调度决策模块;

资源状态监控模块,用于从云计算环境中收集各虚拟机的各项性能指标,并将收集到的信息发送给虚拟机状态预测模块;

虚拟机状态预测模块,用于根据各虚拟机的各项性能指标,并基于状态反馈的虚拟机状态预测模型计算各个虚拟机在此后一段时间内的状态值,并将计算出来的各个虚拟机的状态预测结果发送给调度决策模块;

调度决策模块,用于比较所有虚拟机的状态预测结果,根据比较情况结合请求的任务类型选取要进行资源调度的虚拟机发送给资源调度模块;

资源调度模块,用于将任务调度给调度决策模块选取的虚拟机,实现资源的智能分配。

2. 如权利要求1所述的云计算动态资源调度系统,其特征在于,所述请求的任务类型包括:计算类、I/O类或复合类;所述计算类任务是指以消耗CPU资源为主的任务;所述I/O类任务是指以消耗磁盘I/O或者网络带宽为主的任务;所述复合类任务是指同时消耗多种资源并无法区分哪种资源为该类型任务消耗的主要资源的任务。

3. 如权利要求2所述的云计算动态资源调度系统,其特征在于,所述任务类型判断模块判断请求的任务类型的方法为:

若请求的任务需要的CPU时长比需要的I/O时长大20%以上,则判定该请求的任务为计算类任务;若请求的任务需要的I/O时长比需要的CPU时长大20%以上,则判定该请求的任务为I/O类任务;若请求的任务需要的I/O时长与需要的CPU时长之差小于20%,则判定该请求的任务为复合类任务。

4. 如权利要求1所述的云计算动态资源调度系统,其特征在于,所述虚拟机的各项性能指标包括:CPU利用率、内存利用率、磁盘利用率、磁盘I/O利用率、网络带宽利用率。

5. 如权利要求1-4任意一项所述的云计算动态资源调度系统,其特征在于,所述调度决策模块比较所有虚拟机的状态预测结果,根据比较情况结合请求的任务类型选取要进行资源调度的虚拟机的的方法是:

比较各个虚拟机的状态预测值,如果只有一个虚拟机的预测值最大,则将其选取为要进行资源调度的虚拟机;如果有两个或以上的虚拟机的预测值最大,则根据请求的任务类型计算这些虚拟机的相应负载,在其中选取负载最小的虚拟机作为要进行资源调度的虚拟机。

6. 云计算动态资源调度方法,其特征在于,包括以下实现步骤:

a. 当用户提交任务请求时,任务请求模块验证用户身份的合法性和请求资源的合法性,若验证未通过,则拒绝该任务请求,结束流程;若验证通过,则将该任务请求发送给任务类型判断模块,进入步骤b;

b. 任务类型判断模块对该请求的任务类型进行判断,并将判断出来的任务类型发送给调度决策模块;

c. 虚拟机状态预测模块根据任务类型并结合资源状态监控模块从云计算环境中收集各虚拟机的各项性能指标计算各个虚拟机在此后一段时间内的状态值,并将计算出来的各个虚拟机的状态预测结果发送给调度决策模块;

d. 调度决策模块比较所有虚拟机的状态预测结果,根据比较情况结合请求的任务类型选取要进行资源调度的虚拟机发送给资源调度模块;

e. 资源调度模块将用户请求的任务调度给调度决策模块选取的虚拟机,实现资源的智能分配。

7. 如权利要求6所述的云计算动态资源调度方法,其特征在于,步骤b中,所述请求的任务类型包括:计算类、I/O类或复合类;所述计算类任务是指以消耗CPU资源为主的任务;所述I/O类任务是指以消耗磁盘I/O或者网络带宽为主的任务;所述复合类任务是指同时消耗多种资源并无法区分哪种资源为该类型任务消耗的主要资源的任务。

8. 如权利要求7所述的云计算动态资源调度方法,其特征在于,步骤b中,所述任务类型判断模块对该请求的任务类型进行判断的方法为:

若请求的任务需要的CPU时长比需要的I/O时长大20%以上,则判定该请求的任务为计算类任务;若请求的任务需要的I/O时长比需要的CPU时长大20%以上,则判定该请求的任务为I/O类任务;若请求的任务需要的I/O时长与需要的CPU时长之差小于20%,则判定该请求的任务为复合类任务。

9. 如权利要求6所述的云计算动态资源调度方法,其特征在于,步骤c中,所述各虚拟机的各项性能指标包括:

CPU利用率、内存利用率、磁盘利用率、磁盘I/O利用率、网络带宽利用率。

10. 如权利要求6-9任意一项所述的云计算动态资源调度方法,其特征在于,步骤d中,所述调度决策模块比较所有虚拟机的状态预测结果,根据比较情况结合请求的任务类型选取要进行资源调度的虚拟机的方法为:

比较各个虚拟机的状态预测值,如果只有一个虚拟机的预测值最大,则将其选取为要进行资源调度的虚拟机;如果有两个或以上的虚拟机的预测值最大,则根据请求的任务类型计算这些虚拟机的相应负载,在其中选取负载最小的虚拟机作为要进行资源调度的虚拟机。

云计算动态资源调度系统和方法

技术领域

[0001] 本发明涉及云计算领域中的动态资源调度技术,特别涉及一种基于反馈和预测机制的云计算动态资源调度系统和相应的调度方法。

背景技术

[0002] 近年来,随着信息技术的高速发展和互联网规模的日渐增大,互联网所要处理的业务量和数据量也在迅速增长。为了有效地处理这些海量的数据和服务,优化用户使用互联网服务的体验,云计算技术应运而生。云计算通过分布式计算技术、并行计算技术、虚拟化技术和负载均衡等计算机和网络技术向用户提供方便、快捷、安全的数据存储和网络服务,为计算机技术和 IT 技术的发展带来了新的机遇和挑战。

[0003] 资源调度是指在特定的资源环境下,根据一定的资源使用规则,在不同的资源使用者之间进行资源调整的过程。这些资源使用者对应着不同的计算任务(例如一个虚拟解决方案),每个计算任务在操作系统中对应于一个或者多个进程。通常有两种途径可以实现计算任务的资源调度:在计算任务所在的机器上调整分配给它的资源使用量,或者将计算任务转移到其他机器上。虚拟机的出现使得所有的计算任务都被封装在一个虚拟机内部。由于虚拟机具有隔离特性,因此可以采用虚拟机的动态迁移方案来达到计算任务迁移的目的。

[0004] 虽然云计算是在分布式计算技术、并行计算技术等传统的计算技术的基础上发展起来的,但是相对于传统的分布式计算、并行计算来说,云计算的资源池一般是由一些专门的服务器事先组成好的,并且云计算面向的用户类型和种类众多,因此,一些传统的资源调度和管理技术在云计算环境中并不适用。另外,云计算资源调度还具有按需分配、按量计费、资源流动大、必须对异构环境提供支持、必须降耗节能等新特性,云计算资源调度的这些新特性给云中的资源调度问题带来了更多新的挑战。如何采用高效的资源调度技术较好地解决云计算环境中资源调度的问题,提高云计算中的资源利用率是目前研究的重点。

[0005] 从资源调度的模式上看,传统的分布式计算和并行计算的资源调度主要有三种模式:集中式、分散式、分级式。分级式和分散式的管理模式比较适用于对分布式系统和并行系统进行资源调度。现阶段,在云计算环境中主要是通过虚拟资源池的方式来管理云计算资源,并通过数据中心来实现任务的处理和资源的调度。因此,相对来说,集中式资源管理和调度模式更适用于云计算。目前,关于云计算资源调度的方法的研究已取得了一定的研究成果。澳大利亚的 Rajkumar Buyya 等学者提出的基于经济模型的资源调度方法是当前资源调度的一个主要的方法,该方法通过将云计算环境下的资源供求关系与市场经济模型进行类比,资源的买卖双方通过协商来完成交易,通过价格来调节资源的竞争和分配,从而实现资源的优化分配,并提高效率。徐向华等人在此基础上提出了一种基于市场机制的云计算资源分配策略,并设计了一个基于遗传算法的价格调节算法以平衡市场的需求和供给的矛盾,但是其目前提出的方法仅仅考虑了对 CPU 资源等底层资源的调度问题,还没办法处理其它类型的资源。此外,在基于经济模型的基础上,还衍生出了基于拍卖机制的调度

算法模型。而李建锋等人通过改进现有遗传算法以适应云计算的编程模型框架,提出了一种具有双适应度的遗传算法。Zaharia 等人在 Max-Min 公平调度算法的基础上设计了延迟调度算法,以实现较高的数据本地性,但延迟调度算法在等待开销较大的情况下,会影响任务的完成时间。为此,Isard 等设计了基于最小代价流的调度模型,实现了数据本地性和作业公平性之间的平衡,并应用在了微软的云服务平台。闰英辉提出了一种根据当前各个计算节点资源的使用情况和任务当前的运行状态,将当前云计算环境的整体性能与待调度任务的 QoS 要求相结合,提出了一种为待调度任务选择合适资源的灵活的调度方法。此外,Fujimoto N 等人还提出了基于信任机制的资源调度策略,通过在资源调度时综合考虑信任关系指标来提高资源调度的成功率。由于资源调度方法只能在任务到来时根据当时任务的资源需求状态选择最合适的调度策略,而由于任务在执行的过程中其状态在不断地发生着变化,因此任务在执行过程中的对任务进行重新调度也是一个核心的问题。法国的 Jean-Marc Menaud 和 Hien Nguyen Van 等人通过将为任务选择恰当的虚拟机和为虚拟机选择恰当的物理机的问题转化为约束满足问题实现了资源调度的动态优化。Fabien Hermenier 等人通过研究如何分配和迁移虚拟机到物理主机的问题,提出了一种综合考虑重配置计算时间和虚拟机迁移时间的动态资源调度和管理方法 Entropy。而魏贵义等人则提出了一个基于博弈论的资源调度算法,该算法综合考虑了优化和公平两个方面,在实现上先利用动态规划算法来处理单个参与者的独立优化问题,然后再利用进化算法来处理多个参与者的综合优化问题,能够较好地解决资源的动态调度问题,但是这种方法在实现上比较复杂,仅适用于处理一些非常复杂的和动态的问题。

[0006] 总的来说,虽然近几年研究者在云计算资源调度上已经开展了一些研究工作,但是从结果上看,现有的部分云计算资源调度方法都是面向底层物理资源的调度,主要是通过给虚拟资源优化配置物理资源或是通过虚拟机迁移的方法来实现系统负载均衡和提高资源利用率,而这些方法在实现上都需要停止云应用来实现资源动态调度,有一定应用局限性。此外,另外一些其它方法所涉及到的大多数调度模型中所有的客户的待调度任务的类型都是一致的,而实际情况并非如此。由于云计算中业务众多,任务类型也多种多样,如果将所有用户应用程序的资源调度都置于相同的框架之下,所产生的调度结果对用户来说通常并不是最优的。因此,需要从任务类型的角度出发,根据待调度任务的任务类型选择不同的调度策略,从而实现对不同类型任务的高效调度。并且,在现有的云计算资源调度技术中,已有调度策略的片面性,可能会导致物理资源的盲目流转的出现,需要尽量避免。另外,在任务执行的过程中,由于任务的执行会改变任务的资源消耗类型,因而任务的类型也会因此发生变化,这就使得静态的资源分配和调度往往会造成资源的不足或浪费的情况,而人工的动态资源调整又有着明显的滞后性,因此还需要实时地监控任务的执行状态,并通过任务的资源消耗类型智能地判断任务的资源消耗类型,如果任务的资源消耗类型与初始时的任务类型不一致,且在足够长的时间内也将不会一致,并且这种不一致严重地影响到了虚拟机的健康值,则需要在适当的时机对任务进行动态的重新调度以实现均衡使用各种资源(CPU、内存、磁盘、I/O、网络),避免出现单一资源瓶颈,提升虚拟机密度,消除热点,提高业务处理能力,而现有的资源调度方案没有考虑到这一点。

发明内容

[0007] 本发明所要解决的技术问题是：针对现有技术中的云计算资源分配和调度技术的不足之处，提出一种基于反馈和预测机制的云计算动态资源调度系统和相应的调度方法，能够在云计算环境中实现均衡使用各种计算机资源，实现在较小的开销下获得满意的负载均衡，提高系统调度的综合效率。

[0008] 本发明解决上述技术问题所采用的方案是：

[0009] 云计算动态资源调度系统，包括：

[0010] 任务请求模块，用于在用户提交任务请求时，验证用户身份的合法性和请求资源的合法性，若验证未通过，则拒绝该任务请求；若验证通过，则将该任务请求发送给任务类型判断模块；

[0011] 任务类型判断模块，用于在收到任务请求时，判断请求的任务类型，并将所述请求的任务类型发送给调度决策模块；

[0012] 资源状态监控模块，用于从云计算环境中收集各虚拟机的各项性能指标，并将收集到的信息发送给虚拟机状态预测模块；

[0013] 虚拟机状态预测模块，用于根据各虚拟机的各项性能指标，并基于状态反馈的虚拟机状态预测模型计算各个虚拟机在此后一段时间内的状态值，并将计算出来的各个虚拟机的状态预测结果发送给调度决策模块；

[0014] 调度决策模块，用于比较所有虚拟机的状态预测结果，根据比较情况结合请求的任务类型选取要进行资源调度的虚拟机发送给资源调度模块；

[0015] 资源调度模块，用于将任务调度给调度决策模块选取的虚拟机，实现资源的智能分配。

[0016] 具体的，所述请求的任务类型包括：计算类、I/O 类或复合类；所述计算类任务是指以消耗 CPU 资源为主的任务；所述 I/O 类任务是指以消耗磁盘 I/O 或者网络带宽为主的任务；所述复合类任务是指同时消耗多种资源并无法区分哪种资源为该类型任务消耗的主要资源的任务。

[0017] 具体的，所述任务类型判断模块判断请求的任务类型的方法为：

[0018] 若请求的任务需要的 CPU 时长比需要的 I/O 时长大 20% 以上，则判定该请求的任务为计算类任务；若请求的任务需要的 I/O 时长比需要的 CPU 时长大 20% 以上，则判定该请求的任务为 I/O 类任务；若请求的任务需要的 I/O 时长与需要的 CPU 时长之差小于 20%，则判定该请求的任务为复合类任务。

[0019] 具体的，所述虚拟机的各项性能指标包括：CPU 利用率、内存利用率、磁盘利用率、磁盘 I/O 利用率、网络带宽利用率。

[0020] 具体的，所述调度决策模块比较所有虚拟机的状态预测结果，根据比较情况结合请求的任务类型选取要进行资源调度的虚拟机的的方法是：

[0021] 比较各个虚拟机的状态预测值，如果只有一个虚拟机的预测值最大，则将其选取为要进行资源调度的虚拟机；如果有两个或以上的虚拟机的预测值最大，则根据请求的任务类型计算这些虚拟机的相应负载，在其中选取负载最小的虚拟机作为要进行资源调度的虚拟机。

[0022] 此外，本发明的另一目的，还在于提出相应的资源调度方法，其包括以下实现步骤：

[0023] a. 当用户提交任务请求时,任务请求模块验证用户身份的合法性和请求资源的合法性,若验证未通过,则拒绝该任务请求,结束流程;若验证通过,则将该任务请求发送给任务类型判断模块,进入步骤 b;

[0024] b. 任务类型判断模块对该请求的任务类型进行判断,并将判断出来的任务类型发送给调度决策模块;

[0025] c. 虚拟机状态预测模块根据资源状态监控模块从云计算环境中收集各虚拟机的各项性能指标计算各个虚拟机在此后一段时间内的状态值,并将计算出来的各个虚拟机的状态预测结果发送给调度决策模块;

[0026] d. 调度决策模块比较所有虚拟机的状态预测结果,根据比较情况结合请求的任务类型选取要进行资源调度的虚拟机发送给资源调度模块;

[0027] e. 资源调度模块将用户请求的任务调度给调度决策模块选取的虚拟机,实现资源的智能分配。

[0028] 具体的,步骤 b 中,所述请求的任务类型包括:计算类、I/O 类或复合类;所述计算类任务是指以消耗 CPU 资源为主的任务;所述 I/O 类任务是指以消耗磁盘 I/O 或者网络带宽为主的任务;所述复合类任务是指同时消耗多种资源并无法区分哪种资源为该类型任务消耗的主要资源的任务。

[0029] 具体的,步骤 b 中,所述任务类型判断模块对该请求的任务类型进行判断的方法为:

[0030] 若请求的任务需要的 CPU 时长比需要的 I/O 时长大 20% 以上,则判定该请求的任务为计算类任务;若请求的任务需要的 I/O 时长比需要的 CPU 时长大 20% 以上,则判定该请求的任务为 I/O 类任务;若请求的任务需要的 I/O 时长与需要的 CPU 时长之差小于 20%,则判定该请求的任务为复合类任务。

[0031] 具体的,步骤 c 中,所述各虚拟机的各项性能指标包括:

[0032] CPU 利用率、内存利用率、磁盘利用率、磁盘 I/O 利用率、网络带宽利用率。

[0033] 具体的,步骤 d 中,所述调度决策模块比较所有虚拟机的状态预测结果,根据比较情况结合请求的任务类型选取要进行资源调度的虚拟机的方法为:

[0034] 比较各个虚拟机的状态预测值,如果只有一个虚拟机的预测值最大,则将其选取为要进行资源调度的虚拟机;如果有两个或以上的虚拟机的预测值最大,则根据请求的任务类型计算这些虚拟机的相应负载,在其中选取负载最小的虚拟机作为要进行资源调度的虚拟机。

[0035] 本发明的有益效果是:

[0036] (1) 可以根据待调度任务的任务类型、虚拟机当前的运行状态和基于状态反馈的虚拟机状态预测模型,有倾向性和选择性地资源分配和调度,优化云计算资源的动态调度,使得整个系统资源分配更加合理;

[0037] (2) 在进行状态预测时,将上一次的预测结果作为一个参数反馈到下一轮的预测中去,从而实现对云计算系统中虚拟机状态进行动态的预测,并根据预测值对资源实现动态的调度,从而实现了资源的智能和动态分配,能够在云计算环境中实现均衡使用各种计算机资源,实现在较小的开销下获得满意的负载均衡,提高系统调度的综合效率。

附图说明

[0038] 图 1 为本发明中的调度系统示意图；

[0039] 图 2 为本发明中的调度方法实施例流程图。

具体实施方式

[0040] 如图 1 所示,本发明中的云计算动态资源调度系统包括任务请求模块、任务类型判断模块、资源状态监控模块、虚拟机状态预测模块、调度决策模块、资源调度模块；

[0041] 任务请求模块,用于在用户提交任务请求时,验证用户身份的合法性和请求资源的合法性,若验证未通过,则拒绝该任务请求;若验证通过,则将该任务请求发送给任务类型判断模块；

[0042] 任务类型判断模块,用于在收到任务请求时,判断请求的任务类型,并将所述请求的任务类型发送给调度决策模块；

[0043] 资源状态监控模块,用于从云计算环境中收集各虚拟机的各项性能指标,并将收集到的信息发送给虚拟机状态预测模块；

[0044] 虚拟机状态预测模块,用于根据各虚拟机的各项性能指标,并基于状态反馈的虚拟机状态预测模型计算各个虚拟机在此后一段时间内的状态值,并将计算出来的各个虚拟机的状态预测结果发送给调度决策模块；

[0045] 调度决策模块,用于比较所有虚拟机的状态预测结果,根据比较情况结合请求的任务类型结合请求的任务类型选取要进行资源调度的虚拟机发送给资源调度模块；

[0046] 资源调度模块,用于将任务调度给调度决策模块选取的虚拟机,实现资源的智能分配。

[0047] 由于云计算环境中资源的类型多种多样,各种任务对不同资源的需求也各不相同,各个虚拟机的运行状态也不一样,为了将任务分配给最合适的虚拟机,同时实现根据任务需求的云计算动态资源调度,所以在进行资源调度前需要对任务进行分类,并根据各虚拟机的性能指标对接下来一段时间内所有虚拟机的运行状态进行预测,将虚拟机接下来的运行状态抽象预测为可以量化的状态值,再结合预测的状态值和任务类型选取最适合的虚拟机进行任务分配,从而实现动态资源调度。

[0048] 下面以一个实施例对本发明中的云计算动态资源调度方法进行具体阐述。

[0049] 如图 2 所示,在本例中,实现动态资源调度的步骤如下：

[0050] 资源监控模块实时地监测虚拟机的各项性能指标。如通过资源监控软件或模块对虚拟机的各项性能指标,包括 CPU 利用率、内存利用率、磁盘利用率、磁盘 I/O 利用率、网络带宽利用率等。

[0051] 步骤 1:在任务请求到来时,任务请求模块验证用户身份的合法性和请求资源的合法性,如果验证没有通过,将会拒绝任务请求,而通过验证的任务请求将会继续执行步骤 2；

[0052] 步骤 2:任务类型判断模块智能判断任务请求的类型,常见的任务类型有：

[0053] (1) 计算类任务,此类型任务以消耗 CPU 资源为主,如果任务请求所需要的 CPU 时长明显长于 I/O 时长(CPU 时长比 I/O 时长大 20%以上)则说明该任务是计算类任务；

[0054] (2) I/O 类任务,此类任务主要消耗磁盘 I/O 或网络带宽,如果任务请求所需要的

I/O 时长明显长于 CPU 时长 (I/O 时长比 CPU 时长大 20% 以上) 则说明该任务是 I/O 类任务 ;

[0055] (3) 复合类任务, 此类任务同时消耗多种资源, 很难区分哪种资源是该类型任务消耗的主要资源, 如果任务请求所需要的 I/O 时长和 CPU 时长两者差不太多则说明该任务是复合类任务 ;

[0056] 在任务类型判断模块判断出任务类型后, 执行步骤 3 ;

[0057] 步骤 3 : 根据从云计算环境中收集到的各虚拟机的运行状态和负载信息 (CPU 利用率、内存利用率、磁盘利用率、磁盘 I/O 利用率、网络带宽利用率等), 并通过收集到的虚拟机状态指标确定虚拟机状态值, 具体确定方法如下 :

$$[0058] \quad VSD = \sqrt{\sum_{i=1}^5 (w_i s_i^2)} \quad (\text{式 } 1)$$

[0059] 其中 VSD 代表虚拟机状态值, w_i 代表权值, s_i 为虚拟机状态, 它们分别为 CPU 利用率、内存利用率、磁盘利用率、磁盘 I/O 利用率、网络带宽利用率。将计算出来的 VSD 值添加到历史虚拟机状态值序列的末尾 (如果序列不存在则创建该序列), 并将其长度增加 1。如果历史虚拟机状态值序列大于 n (根据情况选择 n 的值, 此处选为 7), 则删除掉历史虚拟机状态值序列中的第一个值, 跳转到步骤 4, 否则直接跳转到步骤 4 ;

[0060] 步骤 4 : 根据任务的类型和监测到的虚拟机当前各项性能指标以及虚拟机状态预测模块预测虚拟主机列表中各虚拟主机在接下来一段时间内的状态值。预测方法如下 :

[0061] 假设 d_0 是最近的 n 个历史虚拟机状态值序列 (根据情况选择 n 的值, 此处选为 7) :

$$[0062] \quad d_0 = (d_0(1), d_0(2), \dots, d_0(n)) \quad (\text{式 } 2)$$

[0063] 其中, d_1 为 d_0 中数据的累加所生成的 :

$$[0064] \quad d_1 = (d_1(1), d_1(2), \dots, d_1(n)) \quad (\text{式 } 3)$$

$$[0065] \quad d_1 \text{ 与 } d_0 \text{ 的关系为 } d_1(k) = \sum_{i=1}^k d_0(i)$$

[0066] 根据数学理论, 由于累加生成的序列 d_1 的变化趋势为指数趋势, 而一阶微分方程的解恰为指数增长形式的解, 因此可近似认为 d_1 满足一阶微分方程模型 :

$$[0067] \quad \frac{dd_1}{dt} + ad_1 = b \quad (\text{式 } 4)$$

[0068] 其中, d_1 为生成数列, a 和 b 为待求参数, 解此微分方程。由于采用离散形式易于计算, 可将微分方程改写为离散形式 :

$$[0069] \quad a_1[d_1(k+1)] + \frac{1}{2}a[d_1(k+1) + d_0(k)] = b$$

[0070] 其中, a_1 表示了前后顺序增量比关系。

[0071] 由于 $d_1(k) = \sum_{i=1}^k d_0(i)$ 可推断 d_1 与 d_0 之间的关系 :

[0072] 当 $K = 1$ 时, $d_1(k+1) = d_1(2) = d_1(1) + d_0(2)$, 则

$$[0073] \quad d_0(2) = d_1(2) - d_1(1) = d_1(k+1) - d_1(k) = a_1[d_1(k+1)] \quad (\text{式 } 5)$$

[0074] 由此可知,当 $K = 1$ 时, $d_0(2) + \frac{1}{2}a[d_1(1) + d_1(2)] = b$

[0075] 当 $K = 2$ 时, $d_0(3) + \frac{1}{2}a[d_1(2) + d_1(3)] = b$

[0076] 当 $K = n-1$ 时, $d_0(n) + \frac{1}{2}a[d_1(n-1) + d_1(n)] = b$

[0077] 写成矩阵形式有 $\begin{bmatrix} d_0(2) \\ d_0(3) \\ \vdots \\ d_0(n) \end{bmatrix} = \begin{bmatrix} -\frac{1}{2}[d_1(1) + d_1(0)] & 1 \\ -\frac{1}{2}[d_1(3) + d_1(2)] & 1 \\ \vdots & \vdots \\ -\frac{1}{2}[d_1(n) + d_1(n-1)] & 1 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix}$, 简记为 $Y_n = BA$, 由于待求变

量只有 a 和 b 两个, 方程却有 $n-1$ 个, 因此方程无解, 可用最小二乘法得到近似解, 改写为

$$Y_n = B\hat{A} + E, \text{ 为使误差 } E \text{ 最小, 有 } \hat{A} = (B^T B)^{-1} B^T Y_n = \begin{bmatrix} \hat{a} \\ \hat{b} \end{bmatrix}$$

[0078] 将 \hat{a} 和 \hat{b} 代回原方程, 有

$$[0079] \quad \frac{dd_1}{dt} + \hat{a}d_1 = \hat{b} \quad (\text{式 } 6)$$

[0080] 解上述方程可得到 $d_1(t) = \left[d_1(1) - \frac{\hat{b}}{\hat{a}} \right] e^{-\hat{a}t} + \frac{\hat{b}}{\hat{a}}$, 将上式改为离散形式, 又因为

$d_1(1) = d_0(1)$, 因此可以得到

$$[0081] \quad \hat{d}_1(k+1) = \left[d_0(1) - \frac{\hat{b}}{\hat{a}} \right] e^{-\hat{a}k} + \frac{\hat{b}}{\hat{a}} \text{ 其中, } k = 0, 1, \dots$$

$$[0082] \quad \text{根据递推关系, 有 } \hat{d}_1(k) = \left[d_0(1) - \frac{\hat{b}}{\hat{a}} \right] e^{-\hat{a}(k-1)} + \frac{\hat{b}}{\hat{a}} \text{ 其中, } k = 2, 3, \dots$$

[0083] 上式就是时间与资源负载均衡之间的关系预测式, $\hat{d}_1(k)$ 表示第 k 次资源负载均衡值。

[0084] 步骤 5: 调度决策模块负责比较所有虚拟机的预测状态值, 如果只有一个虚拟机的预测状态值最大, 则将其选为要进行资源调度的虚拟机, 如果有两个或两个以上虚拟机的预测状态值都是最大的, 则根据任务类型做进一步的判断:

[0085] 如果任务类型为计算类任务, 则执行步骤 5.1; 如果任务类型为 I/O 类任务, 则执行步骤 5.2; 如果任务类型为复合类任务, 则执行步骤 5.3;

[0086] 步骤 5.1: 当任务为计算类任务时, 调度决策模块计算这些预测状态值最大的虚拟机的 CPU 负载率, 并将其中 CPU 负载率较小的虚拟机作为选定的虚拟机, 若仍有两个或两个以上相同, 则选定最前面的那一台虚拟机。CPU 负载率计算公式为 (其中, M 为 CPU 负

载率, P 为 CPU 利用率, Q 为 CPU 空闲率): $M = P/Q$, 然后执行步骤 6;

[0087] 步骤 5.2: 当任务为 I/O 类任务时, 调度决策模块计算这些预测状态值最大的虚拟机的 I/O 负载率, 并将 I/O 负载率较小的虚拟机作为选定的虚拟机, 若仍有两个或两个以上相同, 则选定最前面的那一台虚拟机。I/O 负载率计算公式为 (其中, N 为 I/O 负载率, V 为磁盘 I/O 利用率, W 为磁盘 I/O 空闲率, X 为网络带宽利用率, Y 为网络带宽空闲率): $N = V/W*50\% + X/Y*50\%$, 然后执行步骤 6;

[0088] 步骤 5.3: 当任务为复合类任务时, 调度决策模块计算这些预测状态值最大的虚拟机的综合负载率, 并将综合负载率较小的虚拟机作为选定的虚拟机, 若仍有两个或两个以上相同, 则选定最前面的那一台虚拟机。综合负载率计算公式为 (其中, O 为综合负载率, P 为 CPU 利用率, Q 为 CPU 空闲率, R 为内存利用率, S 为内存空闲率, T 为磁盘利用率, U 为磁盘空闲率, V 为磁盘 I/O 利用率, W 为磁盘 I/O 空闲率, X 为网络带宽利用率, Y 为网络带宽空闲率): $O = P/Q*20\% + R/S*20\% + T/U*20\% + V/W*20\% + X/Y*20\%$, 然后执行步骤 6;

[0089] 步骤 6: 资源调度模块将任务调度给调度决策模块选取出的虚拟机, 实现资源的智能分配, 至此就完成了一次资源调度的过程。

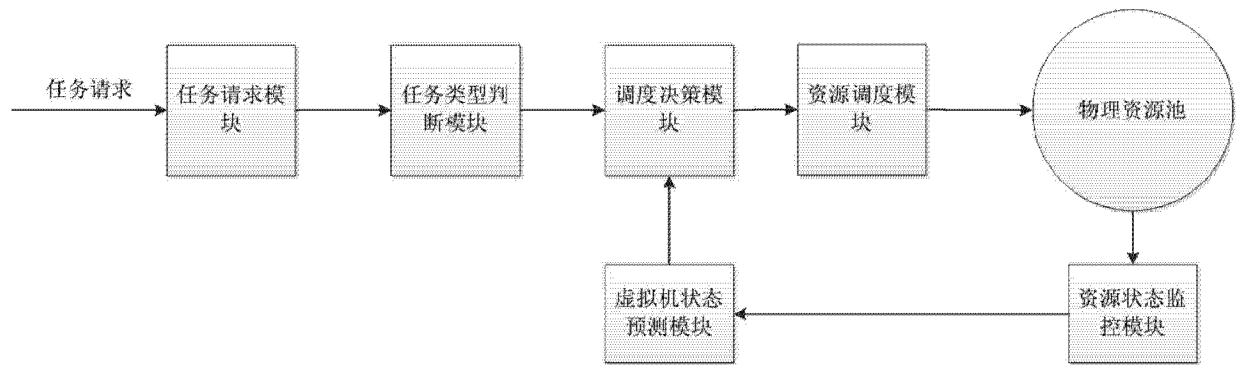


图 1

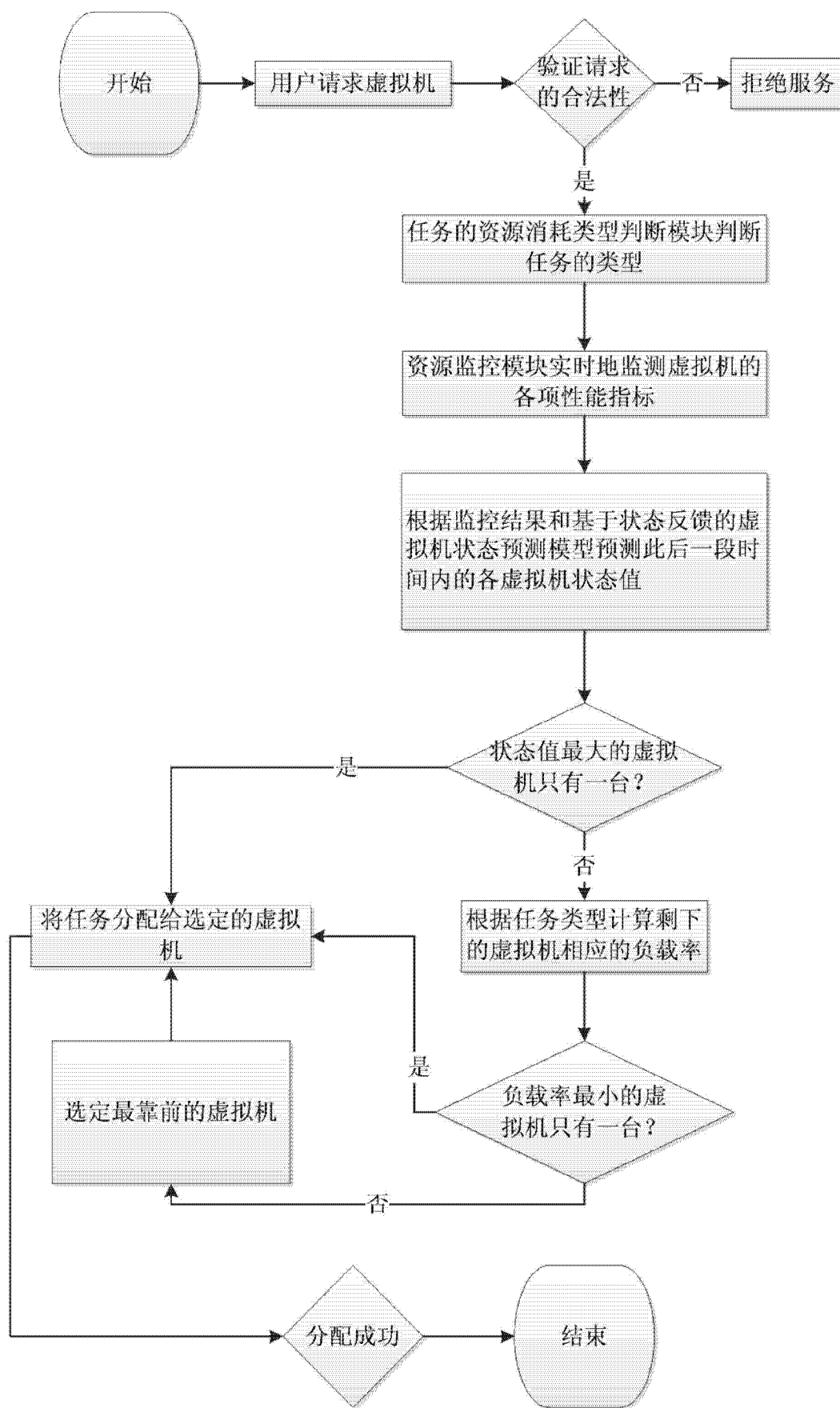


图 2